BlackBerry Java Development Environment

Version: 6.0 beta

Release Notes

BlackBerry

Published: 2010-03-31
SWD-1055591-0331101203-001

Contents

1 Overview..... 2

2 New in this beta release..... 3

3 Fixed issues..... 8

4 Known issues..... 9

5 Legal notice..... 12

Overview

1

BlackBerry® APIs provide you with the classes and interfaces that you can use to create a custom application with features that are designed specifically for a BlackBerry device. You can use the BlackBerry APIs to create applications that perform the following functions:

- interact with specific built-in features of a BlackBerry device, such as the GPS and mapping functions, the camera and video recorder, the phone, and peripherals that use Bluetooth® technology
- respond to interaction with the input methods on the BlackBerry device, such as the trackball, touch screen, or keyboard
- integrate with BlackBerry® Device Software applications, such as the contacts application, the messages application, the BlackBerry® Browser, and BlackBerry® Maps
- interact with the components of the BlackBerry solution, including the BlackBerry® Enterprise Server, push services, and server-side resources

New in this beta release

2

This section lists new features for the BlackBerry® Java® SDK version 6.0 bundle 18.

Feature	Description
new table and list APIs	You can use new table and list APIs to generate dynamic tables and lists in your application. You can use the `TableModel` class to create and modify a table during application runtime. You can create a new table, and add, modify or delete rows, colums and their content in response to user input or events in your application. You can use the `RichTableModel` class to perform the same functions as the `TableModel` class. The `RichTableModel` class allows you to add other UI fields to table cells. You can use the `ListModel` class to create and modify a list during application runtime. You can create a new list, and add, modify or delete list items in response to user input or events in your application. The `ListModel` class also supports patterned list items. You can use the `DataTemplate` class to create list items which follow the same format such as items in an RSS feed.
enhancement to retrieving the default GPS mode	You can use the `GPSInfo.getDefaultGPSMode()` method that is provided in the `net.rim.device.api.gps` package to retrieve the default GPS mode. By default, the `getDefaultGPSMode()` method now returns the modes in the following order: assisted, autonomous, and cellsite. The assisted mode is selected as the default when assisted GPS can be provided to the application without requiring additional credentials from the application. If you do not specify a GPS mode when requesting a location update, the default mode is determined using the GPS capability supported by the wireless network.
new geolocation modes	You can use the following modes that are provided in the new `net.rim.device.api.gps.LocationInfo` class to retrieve location information from the geolocation service: • `GEOLOCATION_MODE` retrieves a location from a geolocation source. The geolocation feature determines and obtains the best location from among the currently available sources based on factors such as network connectivity, location accuracy and data availability. • `GEOLOCATION_MODE_CELL` retrieves a location based on cell tower positioning. • `GEOLOCATION_MODE_WLAN` retrieves a location based on WLAN access point positioning.

Feature	Description
	You can use the `LocationInfo.isModeAvailable()` method that is provided in the `net.rim.device.api.gps.LocationInfo` package to query if a specific mode is available. You can specify a geolocation mode from the `LocationInfo` class, or a mode from the `GPSInfo` class as a parameter for the `LocationInfo.isModeAvailable()` method. You can use the geolocation modes in the `BlackBerryCriteria` class to request a location provider for geolocation from the `BlackBerryLocationProvider` class. If you specify the `GPSInfo.GPS_MODE_CELLSITE` mode, the location is retrieved from cell site towers. If this mode is not supported or available by the wireless service provider, the location is retrieved using `GEOLOCATION_MODE` . For more information about the geolocation modes, visit www.blackberry.com/go/devguides to read the *BlackBerry Java Application Location-Based Services Development Guide*.
turning on and querying Location Services on the device	You can use the `LocationInfo.setLocationOn()` method to turn on the Location Services option on the BlackBerry device. If the Location Services option is turned on, your application can retrieve location information using the available location sources (for example, GPS and geolocation). You should prompt the user before turning on the Location Services option. You can use the `LocationInfo.isLocationOn()` method to determine if the Location Services option on the device is turned on. If Location Services is turned on but the user turns off the geolocation feature in the Location Data option, geolocation services are unavailable.
querying location sources	You can use the following methods in the `LocationInfo` class to determine which location sources (internal GPS receiver, external GPS receiver, or geolocation) are supported or available on the device. • `getSupportedLocationSources()` determines which location sources are supported on the device • `isLocationSourceSupported()` determines if a specified location source is supported on the device • `getAvailableLocationSources()` determines which location sources are available on the device • `isLocationSourceAvailable()` determines if a specified location source is available on the device **Note:** A location source might be supported on the device but unavailable (for example, the user turned the location source off).

Feature	Description
requesting geolocation with GPS updates	You can use one of the following new methods in the `BlackBerryCriteria` class to request geolocation fixes whenever GPS fixes are unavailable. The geolocation fix that is returned is the optimal geolocation fix that is available. • `enableGeolocationWithGPS()` returns a GPS fix as soon as it's available or within the specified timeout period. If the GPS fix is unavailable, a geolocation fix is returned. This method can be used for single and multiple fix requests. • `enableGeolocationWithGPS(intmode)` returns a GPS fix as soon as it's available or within the specified timeout period. With this method, you can specify the option to return the fastest single fix that is available. The fastest single fix might be the first geolocation or GPS fix received.
requesting both GPS and geolocation updates	You can now specify parallel requests for GPS and geolocation updates. For example, you can provide the user with a quick approximate location before a GPS fix is available. To request both GPS and geolocation updates in parallel, create two separate threads to request separate instances of the `BlackBerryLocationProvider` class. One thread specifies a GPS mode, and the other thread specifies a geolocation mode.
foreign key constraint keep support in the SQLite® API	You can use the `DatabaseOptions.set("foreign_key_contraints","on")` method to enable foreign key support for your SQLite database.
support for pinch gestures	Devices with touch screens now support pinch gestures. A pinch begins when two touch points are detected on the touch screen. A pinch's focal point is defined as the midpoint between the two initial touch points. • When two touch points are detected, a `TouchGesture.PINCH_BEGIN` event is sent. The focal point of the pinch gesture is stored as `TouchEvent.getX(1)` and `TouchEvent.getY(1)`. You can access the initial zoom value by invoking `TouchGesture.getPinchMagnitude()`. • a s the user moves one or both touch points, a series of `TouchGesture.PINCH_UPDATE` events are sent. You can access the zoom values during the pinch by invoking `TouchGesture.getPinchMagnitude()`. • hen the user completes the gesture, a `TouchGesture.PINCH_END` event is sent. You can access the final zoom value by invoking `TouchGesture.getPinchMagnitude()`.

Feature	Description
	By default, detection of pinch gestures is not enabled on screens. To enable pinch on a screen, set the screen property `Touchscreensettings.DETECT_PINCH` to 1. You can change the value of this property any time, not just when creating the screen.
enhancements to picture scroll field	You can use the following new methods in the `PictureScrollField` class to enhance the look and feel and functionality of the `PictureScrollField` UI component: • `setForegroundAlpha()` sets focus and unfocus foreground alpha • `setImageDistance()` sets the distance between the vertical centers of two adjacent images. You can also set the image distance to be a negative value to overlap the images. • `setPaddingImageBottom()` sets the padding below the images in pixels • `setScalingRegion()` sets the distance from effects region edge to the center • `setScrollbarVisible()` sets whether or not the scroll bar visible • `setStyle()` sets the Field layout height style • `updateImage()` update the Bitmap image of `ScrollEntry` at index of `ScrollEntry` array. You can now update images at runtime.
Home screen shortcuts	You can create shortcuts and add them to the Home screen on the BlackBerry device by using the new Shortcut API that is provided in the `net.rim.blackberry.api.homescreen` package. When a user clicks a shortcut, the application that is associated with the shortcut opens within a specified context. For example, you can create a shortcut to a contact so that when a user clicks the shortcut, the contacts list application opens and displays the specified contact. You could create shortcuts for a wide variety of reasons. For example, shortcuts to browser bookmarks, map location, or playlists. You can use the following classes and interfaces to create a shortcut: • `ShortcutProvider` provides a factory method for creating shortcuts • `Shortcut` interface that encapsulates a shortcut and specifies information about the shortcut such as the icons to associate with the shortcut and the text to associate with the shortcut • `HomeScreen` provides methods to add shortcuts to the Home screen and check if shortcuts exist • `ShortcutEventListener` interface that provides methods to manage opening, changing, and deleting shortcuts • `Location` specifies where the shortcut should appear on the Home screen

Feature	Description
	• `ShortcutIconField` allows you to provide users with a way to preview a shortcut's appearance on the Home screen from your application's UI • `HomeScreenLocationPicker`, provided in the `net.rim.device.api.ui.picker` package, allows you to create a UI field that users can use to pick where they want the add the shortcut on the Home screen (for example, a folder) **Note:** Some method names in this API will change in the next release.
responding to convenience key events	You can now register your application with the `GlobalKeyProcessRegistry` to receive key events in your application. Your application can now respond to any combination of presses or holds of the convenience keys. Your application must implement a new listener interface, `KeyHandlerRegistryKeyListener`, to handle these events.
new names for classes in the popup menu API	The following name changes in the popup menu API have occured since BlackBerry Java SDK beta 3 bundle 15: • the `CommandItem` class was previously named `ContextItem` • the `CommandItemProvider` class was previously named `ContextItemProvider` The methods in the `Field` class that referred to the previous class names have been renamed also. The following changes have occured: • `getCommandItemProvider()` was previously named `getContextItemProvider()` • `setCommandItemProvider()` was previously named `setContextItemProvider()`
new package for the device capabilities API	The new package name for the device capability API is `net.rim.device.api.system.capability`. This change has been present since BlackBerry Java SDK beta 3 bundle 15. Before that, the `DeviceCapability` class was located in the `net.rim.device.api.ui.capability` package.

Fixed issues

3

This section lists known issues for the BlackBerry® Java® SDK version 6.0 bundle 18.

BlackBerry APIs

The `ContextItemProvider` and `ContextMenuProvider` classes required a RIMAPPSA2 signature. (530838)

Some applications defining custom UI components received runtime exceptions with the message: "Error starting [App Name]: Module '[Module Name]' attempts to access a secure API". (560176)

BlackBerry Integrated Development Environment

BlackBerry® Integrated Development Environment did not support autocomplete for the `LocalEventUpdate` feature's methods and constants. (530520)

BlackBerry Smartphone Simulator

The BlackBerry Smartphone Simulator did not connect to the BlackBerry® MDS Simulator. (540458)

Sample applications

The Message List Demo sample application did not compile. (540256)

Some sample applications did not function as expected.

Affected sample applications and their issues included:

- Camera Demo throws `IllegalArgumentException` then freezes device when launched (540786)
- PIM Demo freezes the device when launched. (540789)
- Spinner Demo cannot select text from `TextSpinBoxField` (534217)

Known issues

4

This section lists known issues for the BlackBerry® Java® SDK version 6.0 bundle 18.

BlackBerry APIs

The classes in the `javax.microedition.m2g` package require a RIMAPPSA2 signature. (528285)

Impact: You cannot run an application that uses any of the above classes on a BlackBerry Smartphone Simulator running in secure mode or a BlackBerry device.

Workaround: Run the application on a BlackBerry Smartphone Simulator in its default insecure mode.

The `TouchGesture.getPinchMagnitude()` method is missing. (559994)

Impact: You cannot retrieve the magnitude of a pinch gesture.

Workaround: None

You cannot create an instance of the `RegionStyles` class in the tables and lists UI API. (559994)

Impact: You cannot apply new styles to table or list regions.

Workaround: None

The following methods in the Home screen shortcut icon API are unavailable: (560192)

- `Shortcut.setIsEditable(boolean)`
- `Shortcut.setIsFavourite(boolean)`
- `HomeScreen.setNewState(boolean)`

Impact: You cannot perform the functions provided by those methods.

Workaround: None

BlackBerry Integrated Development Environment

The BlackBerry® Integrated Development Environment does not support autocomplete for the `AbsoluteFieldManager` class' methods and constants. (530277)

Impact: The BlackBerry Integrated Development Environment does not display autocomplete suggestions when using this class.

Workaround: You must type out the constants or methods.

The RequestSignatures menu item does not appear if you build projects individually. (540458)

Impact: You cannot request the signatures of individual projects in your workspace.

Workaround: You must request signatures for the whole workspace.

When you attach the debugger to a BlackBerry device, a message appears with a list of missing debug files. When you try to debug an application, the class types for local instances of BlackBerry API classes are listed as "unknown". (545144)

Impact: Debugging may be less helpful.

Workaround: Click "do not show this message again" in the missing debug files dialog.

The BlackBerry Integrated Development Environment does not support autocomplete for any classes in the Convenience Key API. (560211)

Impact: The BlackBerry Integrated Development Environment does not display autocomplete suggestions when using this class.

Workaround: You must type out the constants or methods.

The BlackBerry Integrated Development Environment does not support autocomplete for the following methods in the `net.rim.device.api.ui.component.table` package: (560311)

- `DataController.keyChar(char ch, int status, int time)`
- `DataController.keyControl(char ch, int status, int time)`
- `DataController.keyDown(int keyCode, int time)`
- `DataController.keyRepeat(int keyCode, int time)`
- `DataController.keyStatus(int keyCode, int time)`
- `DataController.keyUp(int keyCode, int time)`
- `DataModel.notifyListeners(DataModelChangeEvent event)`
- `DataTemplate.getView()`

Impact: The BlackBerry Integrated Development Environment does not display autocomplete suggestions when using these methods.

Workaround: You must type out the constants or methods.

BlackBerry Smartphone Simulator

When you run the BlackBerry Smartphone Simulator for the second (or subsequent) time, it throws a `NullPointerException`. (530573)

Impact: The BlackBerry Smartphone Simulator launches with errors on subsequent attempts to run.

Workaround: On the **Debug** menu, click **Go** until the error is cleared.

Sample applications

Some sample applications do not function as expected. (530953)

Affected sample applications and their issues include:

- Location Picker Demo throws `IllegalStateException` upon boot. (534227)
- SVG Forms Demo cannot select apples or oranges check box. (535664)
- UDP Demo a "`ConnectionNotFoundException`: Protocol not found" error occurs upon clicking Send. (534228)

Impact: You might not be able to use these sample applications.

Workaround: None

The MediaPlayerLib project does not compile. (559819)

Impact: You cannot run any sample applications that require the MediaPlayerLib project.

Workaround: None

The Phone Screen Demo and Phone Screen Demo 2 sample applications do not compile. (559864)

Impact: You cannot run any the Phone Screen Demo and Phone Screen Demo 2 sample applications.

Workaround: None

Legal notice

©2010 Research In Motion Limited. All rights reserved. BlackBerry®, RIM®, Research In Motion®, SureType®, SurePress™ and related trademarks, names, and logos are the property of Research In Motion Limited and are registered and/or used in the U.S. and countries around the world.

Bluetooth is a trademark of Bluetooth SIG. Google Maps is a trademark of Google Inc. Java, Java SE, Java ME and JavaScript are trademarks of Sun Microsystems, Inc. SQLite is a trademark of Hipp, Wyrick & Company, Inc. Microsoft, Microsoft DirectX, Windows, Windows Vista, and Windows XP Professional are trademarks of Microsoft Corporation. Pentium and Intel are trademarks of Intel Corporation. Wi-Fi is a trademark of the Wi-Fi Alliance. All other trademarks are the property of their respective owners.

This documentation including all documentation incorporated by reference herein such as documentation provided or made available at www.blackberry.com/go/docs is provided or made accessible "AS IS" and "AS AVAILABLE" and without condition, endorsement, guarantee, representation, or warranty of any kind by Research In Motion Limited and its affiliated companies ("RIM") and RIM assumes no responsibility for any typographical, technical, or other inaccuracies, errors, or omissions in this documentation. In order to protect RIM proprietary and confidential information and/or trade secrets, this documentation may describe some aspects of RIM technology in generalized terms. RIM reserves the right to periodically change information that is contained in this documentation; however, RIM makes no commitment to provide any such changes, updates, enhancements, or other additions to this documentation to you in a timely manner or at all.

This documentation might contain references to third-party sources of information, hardware or software, products or services including components and content such as content protected by copyright and/or third-party web sites (collectively the "Third Party Products and Services"). RIM does not control, and is not responsible for, any Third Party Products and Services including, without limitation the content, accuracy, copyright compliance, compatibility, performance, trustworthiness, legality, decency, links, or any other aspect of Third Party Products and Services. The inclusion of a reference to Third Party Products and Services in this documentation does not imply endorsement by RIM of the Third Party Products and Services or the third party in any way.

EXCEPT TO THE EXTENT SPECIFICALLY PROHIBITED BY APPLICABLE LAW IN YOUR JURISDICTION, ALL CONDITIONS, ENDORSEMENTS, GUARANTEES, REPRESENTATIONS, OR WARRANTIES OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, ANY CONDITIONS, ENDORSEMENTS, GUARANTEES, REPRESENTATIONS OR WARRANTIES OF DURABILITY, FITNESS FOR A PARTICULAR PURPOSE OR USE, MERCHANTABILITY, MERCHANTABLE QUALITY, NON-INFRINGEMENT, SATISFACTORY QUALITY, OR TITLE, OR ARISING FROM A STATUTE OR CUSTOM OR A COURSE OF DEALING OR USAGE OF TRADE, OR RELATED TO THE DOCUMENTATION OR ITS USE, OR PERFORMANCE OR NON-PERFORMANCE OF ANY SOFTWARE, HARDWARE, SERVICE, OR ANY THIRD PARTY PRODUCTS AND SERVICES REFERENCED HEREIN, ARE HEREBY EXCLUDED. YOU MAY ALSO HAVE OTHER RIGHTS THAT VARY BY STATE OR PROVINCE. SOME JURISDICTIONS MAY NOT ALLOW THE EXCLUSION OR LIMITATION OF IMPLIED WARRANTIES AND CONDITIONS. TO THE EXTENT PERMITTED BY LAW, ANY IMPLIED WARRANTIES OR CONDITIONS RELATING TO THE DOCUMENTATION TO THE EXTENT THEY CANNOT BE EXCLUDED AS SET OUT ABOVE, BUT CAN BE LIMITED, ARE HEREBY LIMITED TO NINETY (90) DAYS FROM THE DATE YOU FIRST ACQUIRED THE DOCUMENTATION OR THE ITEM THAT IS THE SUBJECT OF THE CLAIM.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW IN YOUR JURISDICTION, IN NO EVENT SHALL RIM BE LIABLE FOR ANY TYPE OF DAMAGES RELATED TO THIS DOCUMENTATION OR ITS USE, OR PERFORMANCE OR NON-PERFORMANCE OF ANY SOFTWARE, HARDWARE, SERVICE, OR ANY THIRD PARTY PRODUCTS AND SERVICES REFERENCED HEREIN INCLUDING WITHOUT LIMITATION ANY OF THE FOLLOWING DAMAGES: DIRECT, CONSEQUENTIAL, EXEMPLARY, INCIDENTAL, INDIRECT, SPECIAL, PUNITIVE, OR AGGRAVATED DAMAGES, DAMAGES FOR LOSS OF PROFITS OR REVENUES, FAILURE TO REALIZE ANY EXPECTED SAVINGS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, LOSS OF BUSINESS OPPORTUNITY, OR CORRUPTION OR LOSS OF DATA, FAILURES TO TRANSMIT OR RECEIVE ANY DATA, PROBLEMS ASSOCIATED WITH ANY APPLICATIONS USED IN CONJUNCTION WITH RIM PRODUCTS OR SERVICES, DOWNTIME COSTS, LOSS OF THE USE OF RIM PRODUCTS OR SERVICES OR ANY PORTION THEREOF OR OF ANY AIRTIME SERVICES, COST OF SUBSTITUTE GOODS, COSTS OF COVER, FACILITIES OR SERVICES, COST OF CAPITAL, OR OTHER SIMILAR PECUNIARY LOSSES, WHETHER OR NOT SUCH DAMAGES WERE FORESEEN OR UNFORESEEN, AND EVEN IF RIM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW IN YOUR JURISDICTION, RIM SHALL HAVE NO OTHER OBLIGATION, DUTY, OR LIABILITY WHATSOEVER IN CONTRACT, TORT, OR OTHERWISE TO YOU INCLUDING ANY LIABILITY FOR NEGLIGENCE OR STRICT LIABILITY.

THE LIMITATIONS, EXCLUSIONS, AND DISCLAIMERS HEREIN SHALL APPLY: (A) IRRESPECTIVE OF THE NATURE OF THE CAUSE OF ACTION, DEMAND, OR ACTION BY YOU INCLUDING BUT NOT LIMITED TO BREACH OF CONTRACT, NEGLIGENCE, TORT, STRICT LIABILITY OR ANY OTHER LEGAL THEORY AND SHALL SURVIVE A FUNDAMENTAL BREACH OR BREACHES OR THE FAILURE OF THE ESSENTIAL PURPOSE OF THIS AGREEMENT OR OF ANY REMEDY CONTAINED HEREIN; AND (B) TO RIM AND ITS AFFILIATED COMPANIES, THEIR SUCCESSORS, ASSIGNS, AGENTS, SUPPLIERS (INCLUDING AIRTIME SERVICE PROVIDERS), AUTHORIZED RIM DISTRIBUTORS (ALSO INCLUDING AIRTIME SERVICE PROVIDERS) AND THEIR RESPECTIVE DIRECTORS, EMPLOYEES, AND INDEPENDENT CONTRACTORS.

IN ADDITION TO THE LIMITATIONS AND EXCLUSIONS SET OUT ABOVE, IN NO EVENT SHALL ANY DIRECTOR, EMPLOYEE, AGENT, DISTRIBUTOR, SUPPLIER, INDEPENDENT CONTRACTOR OF RIM OR ANY AFFILIATES OF RIM HAVE ANY LIABILITY ARISING FROM OR RELATED TO THE DOCUMENTATION.

Prior to subscribing for, installing, or using any Third Party Products and Services, it is your responsibility to ensure that your airtime service provider has agreed to support all of their features. Some airtime service providers might not offer Internet browsing functionality with a subscription to the BlackBerry® Internet Service. Check with your service provider for availability, roaming arrangements, service plans and features. Installation or use of Third Party Products and Services with RIM's products and services may require one or more patent, trademark, copyright, or other licenses in order to avoid infringement or violation of third party rights. You are solely responsible for determining whether to use Third Party Products and Services and if any third party licenses are required to do so. If required you are responsible for acquiring them. You should not install or use Third Party Products and Services until all necessary licenses have been acquired. Any Third Party Products and Services that are provided with RIM's products and services are provided as a convenience to you and are provided "AS IS" with no express or implied conditions, endorsements, guarantees, representations, or warranties of any kind by RIM and RIM assumes no liability whatsoever, in relation

thereto. Your use of Third Party Products and Services shall be governed by and subject to you agreeing to the terms of separate licenses and other agreements applicable thereto with third parties, except to the extent expressly covered by a license or other agreement with RIM.

Certain features outlined in this documentation require a minimum version of BlackBerry® Enterprise Server, BlackBerry® Desktop Software, and/or BlackBerry® Device Software.

The terms of use of any RIM product or service are set out in a separate license or other agreement with RIM applicable thereto. NOTHING IN THIS DOCUMENTATION IS INTENDED TO SUPERSEDE ANY EXPRESS WRITTEN AGREEMENTS OR WARRANTIES PROVIDED BY RIM FOR PORTIONS OF ANY RIM PRODUCT OR SERVICE OTHER THAN THIS DOCUMENTATION.

Research In Motion Limited
295 Phillip Street
Waterloo, ON N2L 3W8
Canada

Research In Motion UK Limited
Centrum House
36 Station Road
Egham, Surrey TW20 9LF
United Kingdom

Published in Canada